

Machine Learning Engineer Nanodegree

Capstone Proposal

Francesco Deleo June 15th, 2020

Proposal

Domain Background

How many Ben & Jerry's tubes or Jack Daniel's BBQ sauce bottles will a store sell next month? Forecasting demand is key for any business selling products and services. Without a solid idea of how sales numbers are going to look like, key business tasks such as managing inventory, predicting cash flows and planning for growth become difficult. In that sense, forecasting demand can help take more informed and intelligent business decisions. Is company X expecting a boom in sales of products and services? Then, X's managers might consider hiring additional staff to successfully meet demand. Are the experts at company Y forecasting a shortfall in sales? Then it would be beneficial to start reducing expenses and reorient Y's marketing efforts.

In the specific case of product demand at physical retail stores, forecasting can be useful in order to better manage inventory, run seasonal offers/discount campaigns, reduce waste, set up appropriate service levels and so on. Accurate forecasts are therefore extremely valuable for companies in retail. Demand forecasting is a well active area of research, with companies often backing its efforts to advance forecasting theory and practice.

My personal motivation for taking on this project is to contribute to a real-world business problem and further expand my knowledge in time-series analysis and grow my machine learning toolbox.

Problem Statement

As the Makridakis Open Forecasting Center (MOFC) puts it: "A wrong weather forecast may result in you carrying around an umbrella on a sunny day, while inaccurate business forecasts could result in actual or opportunity losses." With the advancement of machine learning tools for forecasting, one might ask themselves whether these tools perform well in predicting future product demand. Specifically, I'll try to answer the question: "How effective are Recurrent Neural Networks in forecasting product demand?".

This work will therefore employ AWS Sagemaker DeepAR forecasting algorithm to assess Recurrent Neural Networks' ability in forecasting scalar time-series.

Datasets and Inputs

The data is taken from the "[M5 Forecasting – Accuracy](https://www.kaggle.com/c/m5-forecasting-accuracy/data)" (<https://www.kaggle.com/c/m5-forecasting-accuracy/data>) Kaggle competition and describes hierarchical sales data from Walmart, the world's largest company by revenue, for 1,913 days. The data covers stores in three US States (California, Texas, and Wisconsin) and includes item level, department, product categories, and store details. Although the original dataset is made of 3049 unique products across 10 different stores, this work will focus on the products belonging to one of the food departments in one, handpicked, specific store.

Solution Statement

This work will employ AWS Sagemaker DeepAR forecasting algorithm to assess Recurrent Neural Networks' ability in forecasting scalar time-series. Specifically, the algorithm will be employed to forecast daily sales for the 28 days subsequent to the model training's time-window.

Benchmark Model

The DeepAR model will be evaluated against a benchmark model, which will try to forecast units sold for the 28 days in question based on the previous 28-days average.

Evaluation Metrics

The DeepAR model accuracy will be measured through the Mean Absolute Percentage Error (MAPE), which is a measure of prediction accuracy of a forecasting method.

$$M = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right|$$

The DeepAR model's accuracy will then be evaluated against the Benchmark model's through the comparison of the relative MAPE metrics.

Project Design

The project design will be as follows.

This work will approach time series forecasting in a number of steps:

- Loading and exploring the data
- Creating training and test sets of time series
- Formatting data as JSON files and uploading to AWS S3
- Instantiating and training a DeepAR estimator
- Deploying a model and creating a predictor
- Evaluating the predictor

First of all, a subset of the dataset will be retrieved. This will be data related to one of the food departments for a specific, handpicked, store.

Following, an helper function will be created in order to create a pandas serie for each product in the dataframe. The helper function will first cut those time series for products that were not already being sold in the store at the beginning of the observation period. For example, a single time-series (i.e. a single product) could have a serie of trailing 0s at the beginning of the observation period. These null values will need to be taken care of as they could cause bias in the model. The helper function will cut the time-series in order to deal with this issue.

As the time-series in the dataset develop on the column axis (i.e. 1 column = 1 day), the helper function will also extract each row (i.e. each product), transpose it and attach the dates expressed in time-stamps (which will be taken from the calendar.csv data).

The created time series will then be formatted as JSON files and uploaded to AWS S3 in order to be accessible from the DeepAR estimator.

A DeepAR estimator will be instantiated and its hyperparameters will be tuned through a hyperparameter tuner. The tuner will rank the different models (based on different hyperparameters) through the optimization (minimization) of the Root Squared Mean Error.

The best model will then be extracted from hyperparameter-tuning training job and deployed in order to extract predictions for the 28-day test period.

Finally, the model predictions will be evaluated against the benchmark model through the comparison of their relative MAPE.

References

- Kolassa, S. & Siemsen, E. (2014). Demand Forecasting for Managers. New York: Business Expert Press.
- AWS Sagemaker Documentation (s.d.). DeepAR Forecasting Algorithm [Source](https://docs.aws.amazon.com/sagemaker/latest/dg/deepar.html) (<https://docs.aws.amazon.com/sagemaker/latest/dg/deepar.html>)